AD-A055 697 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 12/2

THE DYNAMIC PROGRAMMING APPROACH TO THE MULTICRITERION OPTIMIZA--ETC(U)

MAR 78 K K BOG

UNCLASSIFIED NL

| OF |

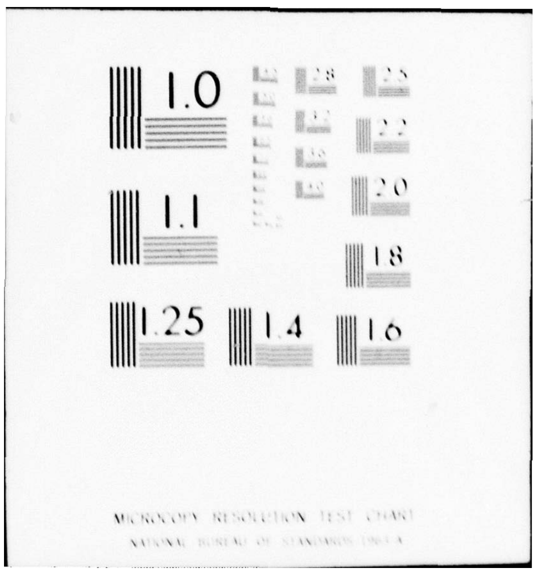
AD
A055 697

END

DATE
FILMED

8 -78

DDC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963-A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

THE DYNAMIC PROGRAMMING APPROACH TO
THE MULTICRITERION OPTIMIZATION PROBLEM

by

Kim Kwang Bog

March 1978

Thesis Advisor: James K. Hartman

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) The Dynamic Programming Approach to the Multicriterion Optimization Problem.		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; March 1978
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kim Kwang Bog		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE March 1978
		13. NUMBER OF PAGES 63 (12) 64p.
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Multicriterion optimization problem; multiple return function; multiple aggregate return function; nondominated solution, decomposition.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Decision makers are often confronted with problems for which there exist several distinct measures of success. Such problems can often be expressed in terms of linear or nonlinear programming models with several "criterion" functions instead of single objective functions. A variety of techniques have been applied to multicriterion problems, but the approach used here, "The Dynamic Programming Approach to Multicriterion Optimization Problem," is based on the concept that the ideal solution to a multiobjective problem — cont

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

25145Ø

aeH

must be a pareto optimal solution. In many cases simply narrowing the set of candidate solutions to the set of all pareto optimal solutions may enable the decision maker to find the compromise being sought. The determination of nondominated points and corresponding nondominated values (pareto optimal solution) related to the multicriterion optimization problem is approached through the use of dynamic programming. The dynamic programming approach has an attractive property which provides the basis for generation of nondominated solutions at each stage by the decomposition method. By using recursive equations we can find out the nondominated points and corresponding nondominated solutions of multiaggregate return function.

Approved for public release; distribution unlimited.

The Dynamic Programming Approach to

the Multicriterion Optimization Problem

by

Kim Kwang Bog
Major, Republic of Korea Marine Corps
B.S., Korea Naval Academy, 1969

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
March 1978

Author Kim Kwang Bog

Approved by: James K Hartman
Thesis Advisor

Gilbert T. Howard
Second Reader

M. G. Sovereign (J K Hartman)
Chairman, Department of Operations Research

D Schrady
Dean of Information and Policy Sciences

ABSTRACT

Decision makers are often confronted with problems for which there exist several distinct measures of success. Such problems can often be expressed in terms of linear or nonlinear programming models with several "criterion" functions instead of single objective functions. A variety of techniques have been applied to multicriterion problems, but the approach used here, "The Dynamic Programming Approach to Multicriterion Optimization Problem," is based on the concept that the ideal solution to a multiobjective problem must be a pareto optimal solution. In many cases simply narrowing the set of candidate solutions to the set of all pareto optimal solutions may enable the decision maker to find the compromise being sought. The determination of nondominated points and corresponding nondominated values (pareto optimal solution) related to the multicriterion optimization problem is approached through the use of dynamic programming. The dynamic programming approach has an attractive property which provides the basis for generation of nondominated solutions at each stage by the decomposition method. By using recursive equations we can find out the nondominated points and corresponding nondominated solutions of multiaggregate return function.

TABLE OF CONTENTS

I. INTRODUCTION

A. PROLOGUE

Situations requiring a decision typically involve multiple conflicting goals which cannot be satisfied simultaneously due either to the inherent conflict among some goals (e.g., maximizing comforts in traveling between two points vs. minimizing cost of travel. Maximizing effectiveness vs. minimizing cost) or to a scarcity of resource (e.g., maximizing productions on each of several individual production lines).

Since first described by Pareto many years ago, the problem of multicriteria, or vector valued optimization has fascinated economists, planners and engineers. Although it has also been considered in a very abstract framework by a number of researchers the most commonly encountered form of the problem seems to be as follows:

$$\text{Maximize } [f_1(x),\ f_2(x), \ldots,\ f_L(x)]$$
$$\text{subject to } g_i(x) \leq 0,\ i\text{-}1,2,\ldots,m$$

where x is a vector.

During the past 10 years increased research in this area has been directed in several directions reflecting different classes of multiple objective decision problems. An excellent recent summary of this area is by K. R. MacCrimmon, [14]. Roy [15] summarized several approaches in 1971 and Johnsen [16] offers an almost encyclopedic coverage of the field to 1975.

A lot of researchers solved the MCO problem by using the concept of a nondominated solution. Yu [17] introduced a general concept of domination structures; the idea helps to view the one-dimensional comparison and the set of nondominated (pareto optimal) solutions as two special cases of a more general solution concept. Zeleny [18] develops a multicriteria simplex method for generating all nondominated extreme points. Polak [19] presents an algorithm which constructs an economical grid of noninferior points to be used in computation with an interpolation scheme in the value space. The algorithm is specifically designed for the bicriteria case, but it can also be used in higher dimensional situations.

The algorithm we shall present in this thesis is specialized to the two or more criteria space with only a few equality or inequality type constraints (only one constraint in our example).

The theory of the dynamic programming approach to the multiple criteria optimization problem [MCODP] is developed by combining the nondominated solution concepts and dynamic programming method.

Using the dynamic programming decomposition theorem, we break the multicriteria problem into N stages. At each stage we find the nondominated solution by a recursion equation and hence solve the MCO problem by finding the nondominated solutions of multi-aggregate return function.

B. MULTICRITERION OPTIMIZATION PROBLEM ANALYSIS - A REVIEW OF THE LITERATURE

$$\text{Maximize } F(x)$$

$$\text{subject to } g_i(x) \leqq 0 \quad \text{for} \quad i=1,2,\ldots,m$$

where x is an N-vector $(x_1, x_2,\ldots,x_N)$, and $F(x)$, a function of x, is an L-vector with scalar components $(f_1(x), f_2(x),\ldots, f_L(x))$. The solution to this problem is the set of all x which are nondominated and which satisfy the constraints $g_i(x) \leqq 0$ for $i=1,2,\ldots,m$.

A solution vector x is nondominated if it satisfied the constraints and if there does not exist another x' also satisfying them and having the properties that

$$f_i(x') \leqq f_i(x) \quad \text{for} \quad i=1,2,\ldots,L$$

$$f_i(x') > f_i(x) \quad \text{for at least one such } i.$$

Although interest in multicriterion problems is quite recent, the literature concerning these problems is already voluminous.

For a bibliography and synthesis of work in this area see MacCrimmon [2] . In his paper, written in 1973, he discussed every method which was developed up to then and attempted to group them into four main categories to facilitate comparisons among the various methods. They are

a) weighting methods

b) sequential elimination methods

c) mathematical programming methods

d) spatial proximity methods.

The class of <u>weighting methods</u> has received the most attention and particular models within this class have been the most widely applied. Although these weighting methods seem very diverse, they all have the following characteristics:

- A process comparing attributes by obtaining numerical scalings of attribute values (intra-attribute preference) and numerical weights across attributes (inter-attribute preference).

MacCrimmon divided the weighting method into nine weighting methods. Some of them are:

linear regression

trade-offs

simple additive weighting.

The linear regression method can be used when the problem is in such situations that are sufficiently repetitive to group together. Attributes considered by the decision maker can be treated as variables in a linear model. From his past choice, coefficients of the attributes are estimated using standard regression techniques.

Slovic and S. Lichtenstein [3] provide a very comprehensive survey of linear regression models and have an excellent bibliography.

The trade-off methods differ from the linear-regression method in that the preference of the decision maker is obtained by directly asking him his preferences rather than by inferring them from his past choices. Although this method has the advantage of obviating the need for a considerable past history of similar situations, it has the disadvantage of possibly finding that the decision maker is unable to verbalize his true preferences.

In the trade-off approach, the marginal rates at which the decision maker is willing to trade one attribute for another are obtained by direct questioning. If all the relevant trade-offs can be made, then the focus of the problem can be narrowed to one attribute. Raiffa [5] utilizes a more general trade-off method on the very complex problem of obtaining certainty equivalents for lotteries in multiple attribute decision problems.

In the simple additive weighting method, for each of the attributes decision problems.

In the simple additive weighting method, for each of the attributes the decision maker assigns importance weights which become the coefficients of the variables. To reflect his marginal worth assessments within attributes, the decision maker also makes a numerical scaling of intra-attribute values. He can then obtain a total score for each alternative simply by multiplying the scale rating for each attribute value by the importance weight assigned to the attribute and then summing these products over all attributes.

After the total scores are computed for each alternative, the alternative with the highest score is the one prescribed to the decision maker. Simple additive weighting is very widely used. One important use of this technique was in the mustering out system of the U. S. Army at the end of World War II [6].

<u>Sequential elimination methods</u> are less demanding of the decision maker than weighting methods. Sequential elimination methods are characterized by a process for sequentially comparing alternatives on the basis of attribute values so that alternatives can be either eliminated or retained.

MacCrimmon distinguished among four methods. One of these is the dominance method which compares one real alternative against another real one to see if proper alternatives can be eliminated. If one alternative has attribute values that are at least as good as those of another alternative for all attributes, and if it has one or more values that are better, then the first alternative is said to "dominate" the second, and the second alternative can be eliminated. Although this is probably the least controversial decision rule, it unfortunately does not often succeed in eliminating very many alternatives.

Terry [7] shows how dominance can be used as an initial filter in choosing among new product areas for comparing diversification. Freiner and Yu [8] combine dominance with the mathematical programming approaches.

The class of <u>programming methods</u> has recently begun to receive much attention. MacCrimmon divided the mathematical programming methods into three methods, linear programming, goal programming and interactive, multi-criterion programming.

A regular linear programming model may be viewed as a multiple attribute decision method. The variables are the attributes. The linear constraints are conjunctive constraints on combinations of attributes, and there is a linear, compensatory objective function.

Efficient solution algorithms are available. The classic diet problem [9] is an example of this method. The objective is to provide a balanced diet, through the selection of the amounts of particular foods, that will satisfy particular nutritional standards at minimum cost.

In a goal programming formulation, the decision maker specifies acceptable or desired levels on single attribute values (i.e., one-

variable constraints) or on combinations of attributes (i.e., multi-variable constraints) and these serve as the primary goals.

Lee and Claytor [10] apply goal programming to the scheduling of an academic department. The attributes (variables) are defined in terms of faculty and non-faculty personnel having varying degree qualifications. The goals involve faculty/student ratios, faculty/staff ratios, percentage increase in salaries, low cost, etc. Solutions obtained under different assumptions about available budget and accreditation requirements are presented.

Ijiri [11] develops the accounting aspects of goal programming.

Interactive, multi-criterion programming does not assume a global objective but rather requires the decision maker to provide his local trade-offs in the neighborhood of a feasible alternative. These trade-offs (on the attributes or criterion involving attributes) are used in a local objective function for a mathematical programming algorithm to generate an optimal solution for that objective. The decision maker then has an opportunity to provide new trade-offs which again serve as inputs to the algorithm. This process continues until the decision maker no longer wishes to revise his trade-offs and so an optimal solution is reached.

Geoffrion, Dyer, and Feinberg [12] describe the use of the method in scheduling an academic department. Related approaches are given in the studies of Benayoun de Montgolfier and Tergny [13] and Yu, Zeleny.

While the preceding three classes are probably the most common general approaches to multi-attribute decision problems, some of the more specialized methods are also receiving attention. Since they all make explicit use of spatial representation, MacCrimmon calls them

"spatial proximity methods." He distinguished among three methods in spatial proximity methods. One of them is the indfiference map method.

The decision maker's preferences can be obtained in the form of indifference surfaces which show the combinations of attribute values that are equally preferred. The alternatives to be considered can be located in the same spatial representation, and by identifying the indifference surface on which they lie, a complete ordering among the alternatives can be generated.

MacCrimmon [14] gives an example of the use of this procedure for transportation system planning. Many combined approaches are possible and hopefully this overview will provide some insight into the available methods so that readers can begin to consider useful combinations for their own multiple objective decision problems.

II. BASIC THEORY OF DYNAMIC PROGRAMMING APPROACH TO MULTICRITERION OPTIMIZATION PROBLEM

A. INTRODUCTION

This chapter is concerned with finding solutions to the following general problem, which will be called "MCO (Multicriterion Optimization)" problem.

Find the set of nondominated solutions to

$$\begin{aligned} \text{maximize} \quad & f_1(x) \\ & f_2(x) \\ & \vdots \\ & f_L(x) \end{aligned}$$

subject to

$$\begin{aligned} Ax &\le b \\ x &\ge 0 \end{aligned}$$

where A is an M by N matrix

b is a vector $(b_1, b_2, \ldots, b_N)$

x is a vector $(x_1, x_2, \ldots, x_N)$

$f_i(x)$ is a decomposable function

The dynamic programming approach to the multiple objective function problem is of interest because we can decompose each objective function into N-stages in dynamic programming (if separability and monotonicity conditions hold). At each stage, we get the L single-variable return functions and we can find the nondominated policy by just eliminating the dominated policy.

Once we get the optimal returns (nondominated solution) from each stage, then we can easily obtain a nondominated solution for a system by Bellman's "principle of optimality" stated more succinctly in his words -

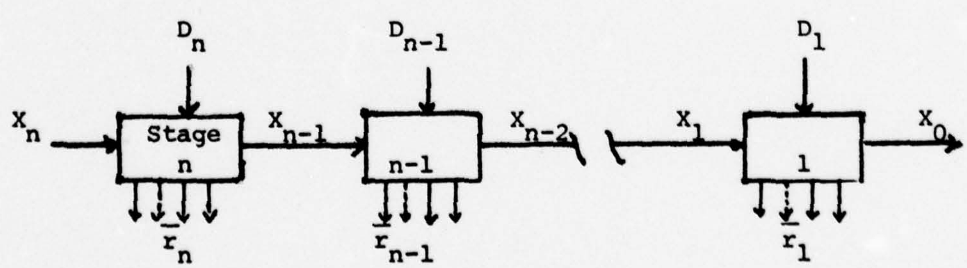
> An optimal policy has the property that whatever the initial state and decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision [20].

B. NOTATION

The following notation and basic definitions will be employed in the remainder of this chapter.

1. Multistage Problem Solving

When solving a complex problem, we often break it into a series of smaller problems - decomposition - and then combine the results from the solution of the whole problem - composition. We call this approach multistage problem solving.



[Figure 1]

2. State (x)

The state of each stage is defined by a set of numbers called state variables

$$x = (x_1, x_2, \ldots, x_N)$$

We use a subscript n to denote the value of the state variables after stage n - see (Fig. 1).

3. Decision Variable (D)

The decision variables are those quantities that can be controlled or chosen in the design and operation of any stage

$$D = (D_1, D_2, \ldots, D_n, \ldots, D_N)$$

See (Fig. 1).

4. Stage Transformation

Each stage of the process transforms the state of its input into an output state in a way dependent on the decision that has been made for the operation of the stage. We express this symbolically by writing

$$x_{n-1} = t_n(X_n, D_n)$$

meaning that given X_n and D_n it is possible to calculate X_{n-1}.

5. Multiple Return Function

Each stage has L return functions. We call this a "multiple return function" and the notation is as follows

$$\overline{r}_n = \begin{bmatrix} r_n^{\,1} \\ r_n^{\,2} \\ \vdots \\ r_n^{\,\ell} \\ \vdots \\ r_n^{\,L} \end{bmatrix}$$

where

$n = 1,2,\ldots,$ N (stages)

$\ell = 1,2,\ldots,$ L (number of return functions)

6. Multiple Aggregate Return Function

The total return R of the ℓ^{th} return from stage one through N is some function of the ℓ^{th} individual stage returns written as

$$\overline{R} = \begin{bmatrix} R^1 \\ R^2 \\ \vdots \\ R^L \end{bmatrix}$$

$$R^{\ell} = g^{\ell}[r_N^{\,\ell},\ r_{N-1}^{\,\ell},\ldots,\ r_1^{\,\ell}]$$

$\ell = 1,2,\ldots,$ L (number of return function)

7. Inequality Notation:

If the objective functions of an optimization problem is vector valued, such that

$$\overline{Q} = [Q^1,\ Q^2,\ldots,\ Q^{\ell}]^T,$$

we define a partial ordering of elements $\overline{Q}$ as follows:

$$\bar{Q}^{o} < \bar{Q}^{*} \leftrightarrow Q^{oi} < Q^{*i} \quad \forall_i$$

$$\bar{Q}^{o} \leq \bar{Q}^{*} \leftrightarrow Q^{oi} \leq Q^{*i} \quad \forall_i$$

$$\text{and } \exists\ j \ni Q^{oj} \neq Q^{*j}$$

$$\bar{Q}^{o} \leq \bar{Q}^{*} \leftrightarrow Q^{oi} \leqq Q^{*i} \quad \forall_i$$

$$\bar{Q}^{o} = \bar{Q}^{*} \leftrightarrow Q^{oi} = Q^{*i} \quad \forall_i$$

8. <u>Notation of Nondominated Points</u>

Given a set of feasible points

$$V \subset R^n$$

and a vector criterion function

$$f : R^n \rightarrow R^P$$

then the set of nondominated points is

$$\Omega = \{x^* \varepsilon\ V \mid f(x^1) \geq f(x^*) => x^1 \notin V\}.$$

9. <u>Notation of Nondominated Values</u>

$$\Omega = \{Z = R^P \mid z = f(x),\ x \varepsilon \Omega\}$$

The set of all nondominated solutions in the criteria space $\bar{T}_N(X_N)$ will be denoted by

$$\underset{D_N,---,D_1}{\bar{\Omega}\ [\bar{T}_N\ (X_N)]}$$

where $\bar{T}_N(X_N) = \{\bar{g}(\bar{r}_N(X_N,D_N),\ \bar{r}_{N-1}(X_{N-1},D_{N-1}),---\bar{r}_1(X_1D_1)),\ \forall\ D_N,\ldots,D_1\}$

that is, $\overline{T}_N(X_N)$ is the set of all feasible values of the vector

$\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1,D_1))$ for all $D_N,---,D_1$,

subject to

$$X_{n-1} = t_n(X_n,D_n),\ n = 1,2,---N$$

and $\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\ \overline{r}_1(X_1,D_1))$ is

the multiple aggregate return function.

The optimal return (nondominated solution) from N stages is defined as

$$\overline{F}_N(X_N) = \underset{D_N,---,D_1}{\overline{\Omega}} [\overline{T}_N(X_N)]$$

subject to

$$X_{n-1} = t_n(X_n,D_n),\ n = 1,2,---N.$$

10. <u>Multiple Recursion Equation</u>

Assuming that the optimal return,

$\overline{F}_N(X_N) = \underset{D_N,---,D_1}{\overline{\Omega}} [\overline{T}_N(X_N)]$ is decomposable,

then

$$\overline{F}_N(X_N) = \underset{D_N}{\overline{\Omega}} [\overline{S}_N(X_N)],$$

where

$$\overline{S}_N(X_N) = \{\overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{F}_{N-1}(X_{N-1})),\ \forall\ D_N\}$$

that is, $\bar{S}_N(X_N)$ is the set of all feasible values of the vector

$\bar{g}_1(\bar{r}_N(X_N,D_N), \underset{D_{N-1},---,\ D_1}{\bar{\Omega}} [\bar{T}_{N-1}(X_{N-1})])$ for all values of D_N.

This notation means that each $\bar{r}_N(X_N,D_N)$ is combined with each vector in

the set $\underset{D_N,---,\ D_1}{\bar{\Omega}} [\bar{T}_{N-1}(X_{N-1})]$

subject to

$$X_{N-1} = t_N(X_N,D_N),$$

and let

$$\bar{Q}_N(X_N,D_N) = \bar{g}_1(\bar{r}_N(X_N,D_N), \bar{F}_{N-1}(X_{N-1}))$$

then

$$\bar{S}_N(X_N) \quad = \{\bar{Q}_N(X_N,D_N), \forall D_N\}.$$

Applying the same decomposition to $\bar{F}_{N-1}(X_{N-1}),---, \bar{F}_2(X_2)$, we obtain the multiple recursion equations.

$$\bar{F}_n(X_n) = \underset{D_n}{\bar{\Omega}} [\bar{S}_n(X_n)] \quad n = 1,---,N$$

where

$$\bar{S}_n(S_n) = \{\bar{Q}_n(X_n,D_n), \forall D_n\}$$

$$\bar{Q}_n(X_n) = \bar{r}_n(X_n,D_n) \quad n = 1$$

$$= \bar{r}_n(X_n,D_n) \circ \bar{F}_{n-1}(X_{n-1},D_{n-1}) \quad n = 2,---,N$$

and "o" is the composition operator

subject to

$$X_{n-1} = t_n(X_n, D_n).$$

C. BASIC CONCEPTS OF "NONDOMINATED"

Let us briefly outline the basic concept. We will utilize the decision variable D_n as in Figure 2. The value of decision variable D_n results in the value of $\bar{Q}_n(X_n, D_n)$ through the equation.

$$D_n \longrightarrow \boxed{\begin{array}{r} \bar{Q}_n(X_n, D_n) = \bar{r}_n(X_n, D_n) \\ \circ\ \bar{F}_{n-1}(t_n(X_n, D_n)) \end{array}} \begin{array}{l} \longrightarrow Q_n^{1}(X_n, D_n) \\ \longrightarrow Q_n^{2}(X_n, D_n) \\ \quad\vdots \\ \longrightarrow Q_n^{L}(X_n, D_n) \end{array}$$

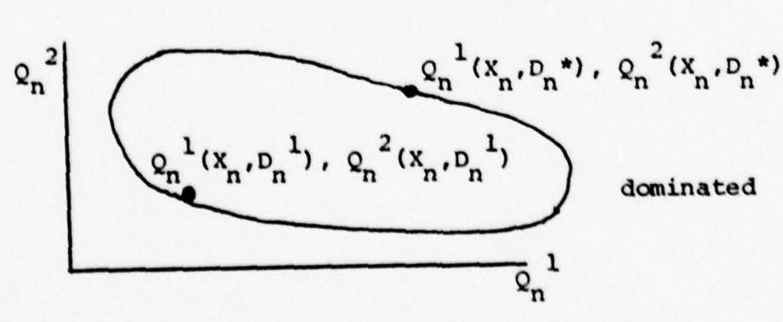
$n = 1, 2, \ldots, N$ (number of stages)

$\ell = 1, 2, \ldots, L$ (number of objective functions)

X_n = fixed.

[Figure 2]. The relationship between decision variables and return functions.

Each choice of decision values which is allowable yields a feasible solution in criteria space as in Figure 3a.



[Figure 3a]. Feasible solutions.

The full set of allowable solutions obtained by mapping all allowable values of the decision variable D_n into the criteria space yields some volume in that space, again as in Figure 3a.

In general, the majority of these feasible solutions will be dominated, that is, a feasible solution will exist which is at least as good in all criteria and better in at least one criterion.

A "nondominated" policy can be defined in the following way. A point $D_n^* \varepsilon X_n$ is said to be a nondominated policy if and only if there is no other policy $D_n^1 \varepsilon X_n$ such that

$$Q_n^i(X_n, D_n^1) \geqq Q_n^i(X_n, D_n^*), \text{ for all } i$$

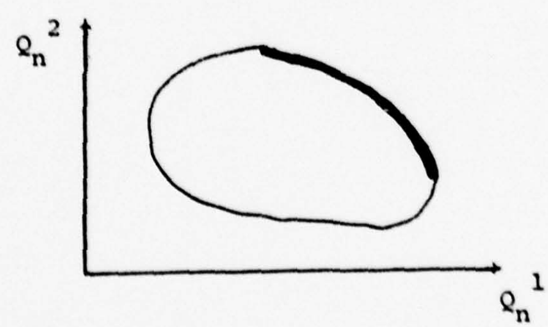
and

$$Q_n^i(X_n, D_n^1) \neq Q_n^i(X_n, D_n^*), \text{ at least one } i$$

where X_n is fixed.

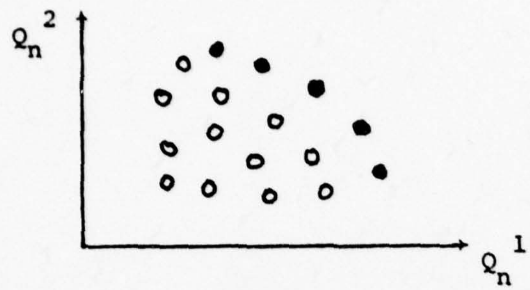
That is, a "nondominated" policy has the property that there is no other feasible policy which would improve at least one goal variable while the others stay unchanged.

The surface of nondominated solutions in criteria space for the example shown is indicated in Figure 3b.



[Figure 3b]. Surface of nondominated solutions.

A set of representative nondominated solutions for the example shown is indicated in Figure 3c.



[Figure 3c]. Representative nondominated solutions.

D. BASIC THEOREM

From the general M C O problem,

$$\begin{array}{ll} \text{maximize} & F(x) \\ \text{subject to} & g(x) \leq 0 \quad \text{for} \quad i = 1,2,\ldots,m \end{array}$$

where x is an N-vector and F(x) a function of x, is an L-vector with scalar component $(f_1(x), f_2(x),\ldots,f_L(x))$.

To use the dynamic programming method each of the L objective functions must satisfy the definition [1] and [2].

1. <u>Definition [1] (Separability)</u>

Let the aggregate return be

$$R = g[r_N, r_{N-1},\ldots, r_1].$$

"Separability" can be written

$$g \; [r_N, r_{N-1},\ldots, r_1]$$

$$= g_1 \; [r_N, g_2(r_{N-1},\ldots,r_1)]$$

where g_1 and g_2 are real-valued functions.

2. Definition [2] (Monotonicity)

From $g_1[r_N, g_2(r_{N-1},\ldots,r_1)]$

g_1 is a monotonically nondecreasing function of g_2 for every fixed r_N. If each objective function satisfies the definition [1], [2], then it can be decomposed into N stages. At each stage we get L return functions.

$$\bar{r}_n(X_n,D_n) = \begin{bmatrix} r_n^{1}(X_n,D_n) \\ r_n^{2}(X_n,D_n) \\ \vdots \\ r_n^{L}(X_n,D_n) \end{bmatrix}$$

$n = 1,2,\ldots,N$ (stages)

$\ell = 1,2,\ldots,L$ (number of return functions)

and

$$\bar{Q}_n(X_n,D_n) = \bar{r}_n(X_n,D_n) \circ \bar{F}_{n-1}(t_n(X_n,D_n))$$

giving

$$\bar{Q}_n(X_n,D_n) = \begin{bmatrix} Q_n^{1}(X_n,D_n) \\ Q_n^{2}(X_n,D_n) \\ \vdots \\ Q_n^{L}(X_n,D_n) \end{bmatrix}$$

We assume that there is some overall utility function implicitly defined on the $\bar{Q}_n(X_n,D_n)$. At stage M X_n is fixed and the feasible

values of D_n depend on X_n.

$$U[\overline{Q}_n(X_n,D_n)] = U[Q_n^1(X_n,D_n),---Q_n^L(X_n,D_n)]$$

This function is unknown but it can be assumed to be real-valued, monotonic and nondecreasing in each argument $Q_n^\ell(X_n,D_n)$.

The problem is to find D_n^* for fixed X_n such that

$$\max_{D_n \varepsilon X_n} U[\overline{Q}_n(X_n,D_n)] = U[\overline{Q}_n(X_n,D_n^*)].$$

3. <u>Theorem [1] [Geoffrion]</u>

Let the set of feasible D_n be compact. Then at least one policy at which $U[\overline{Q}_n(X_n,D_n)]$ achieves its maximum over X_n (fixed) is "non-dominated." That is

$$\max_{D_n \varepsilon X_n} U[\overline{Q}_n(X_n,D_n)] = U[\overline{Q}_n(X_n,D_n^*)]$$

where $D_n^* \varepsilon \Omega$ and Ω = the set of nondominated points.

PROOF

By the compactness (bounded and closed) of the feasible set of D_n and the continuity of U there is at least one maximal solution to $U[\overline{Q}_n(X_n,D_n)]$, say $D_n^* \varepsilon \Omega$. Consider some point $D_n' \varepsilon X_n$ such that

$$Q_n^i(X_n,D_n') \geqq Q_n^i(X_n,D_n^*) \qquad i = 1,2,\ldots,L$$

and which maximizes $\sum_{i=1}^{L} Q_n^i(X_n,D_n)$ over X_n. The point D_n' is obviously nondominated since otherwise there would be some $\overline{D_n} \varepsilon X_n$ with

$$Q_n^i(X_n,\overline{D_n}) > Q_n^i(X_n,D_n') \quad \text{for at least one i}$$

and

$$Q_n^k(X_n,\overline{D_n}) \geqq Q_n^k(X_n,D_n') \text{ for the remaining } k \neq i.$$

Then, of course,

$$\Sigma Q_n^i(X_n,\overline{D_n}) > \Sigma Q_n^i(X_n,D_n') \text{ which contradicts the choice}$$

of D_n'.

Since D_n' is nondominated

$$\overline{Q}_n(X_n,D_n) \geqq \overline{Q}_n(X_n,D_n') \Rightarrow \overline{Q}_n(X_n,D_n) = \overline{Q}_n(X_n,D_n')$$

$$\forall D_n \in X_n$$

Because U is monotone increasing, then also, letting $D_n = D_n^*$

$$\overline{Q}_n(X_n,D_n^*) \geqq \overline{Q}_n(X_n,D_n') \Rightarrow \overline{Q}_n(X_n,D_n^*) = \overline{Q}_n(X_n,D_n')$$

implies

$$U[\overline{Q}_n(X_n,D_n^*)] \geqq U[\overline{Q}_n(X_n,D_n')] \Rightarrow U[\overline{Q}_n(X_n,D_n^*)]$$

$$= U[\overline{Q}_n(X_n,D_n')]$$

so, D_n' also solves

$$\underset{D_n \in X_n}{\text{MAX}}\ U[\overline{Q}_n(X_n,D_n)] = U[\overline{Q}_n(X_n,D_n')]$$

Thus, it is shown that at least one policy at which $U[\overline{Q}_n(X_n,D_n)]$ achieves its maximum over X_n is nondominated

Q.E.D.

The theorem provides the justification for solving for nondominated solutions.

Next, let us briefly explain the theory of the dynamic programming for the MCO problem. In dynamic programming the N-variable problems must be broken into N one-variable problems. So our objective is to decompose the problem

$$\overline{F}_N(X_N) = \overline{\Omega}_{D_N,---,D_1} [\overline{T}_N(X_N)]$$

where $\overline{T}_N(X_N) = \{\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---\overline{r}_{.}(X_1,D_1)),\ \forall\ D_N,---D_1\}$

that is, $\overline{T}_N(X_N)$ is the set of all feasible values of

$\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1,D_1))$ for all $D_N,---D_1$

subject to

$$X_{n-1} = t_n(X_n,D_n),\quad n=1,2,---,N$$

into N equivalent subproblems each containing only one state variable and one decision variable. Then the solutions from the subproblems are combined to obtain the solution to the original problem. To achieve this decomposition, a highly restrictive assumption must be made about the function $\overline{g}_i$. That is, a sufficient condition for decomposition is that $\overline{g}_2$ must be a monotonically nondecreasing function and another condition for the decomposition is the separability.

In the case of a single objective function, a sufficient condition for achieving the decomposition has been given by Mitten [33]. We will prove that a condition similar to Mitten's is a sufficient condition for the case of multiple objective function in theorem [2], [3], [4].

To decompose the problem

$$\overline{F}_N(X_N) = \overline{\Omega}_{D_N,---,D_1} [\overline{T}_N(X_N)]$$

where $\overline{T}_N(X_N) = \{\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1,D_1)),\ \forall\ D_N,---D_1\}$

we must achieve the crucial step of moving the optimization (here finding all nondominated solutions) with respect to $D_{N-1},---,D_1$ inside the Nth stage return. Let the multiple objective function be separable as follows:

$$\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1,D_1))$$

$$= \overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{g}_2(\overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1,D_1)))$$

then

$$\overline{T}_N(X_N) = \{\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{r}_{N-1}(X_{N-1},D_{N-1}),---\ \overline{r}_1(X_1,D_1)),\ \forall\ D_N,---D_1\}$$

$$= \{\overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{g}_2(\overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1 D_1))),$$

$$\forall\ D_N,---D_1\}.$$

From the definition of $\overline{F}_N(X_N)$ it follows that

$$\overline{F}_{N-1}(X_{N-1}) = \underset{D_{N-1},---,D_1}{\overline{\Omega}}\ [\overline{T}_{N-1}(X_{N-1})]$$

where

$$\overline{T}_{N-1}(X_{N-1}) = \overline{g}_2(\overline{r}_{N-1}(X_{N-1},D_{N-1}),---,\overline{r}_1(X_1,D_1)),\ \forall\ D_{N-1},---D_1\ .$$

Then the decomposition is as follows:

$$\overline{F}_N(X_N) = \underset{D_N}{\overline{\Omega}}\ [\overline{S}_N(X_N)]$$

where

$$\overline{S}_N(X_N) = \{\overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{F}_{N-1}(X_{N-1})),\ \forall\ D_N\}$$

$$= \{\overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{F}_{N-1}\ t_N(X_N,D_N)\ \forall\ D_N\}.$$

It is clear that we can proceed further by treating $\overline{F}_{N-1}(X_{N-1})$ and then $\overline{F}_{N-2}(X_{N-2}),\text{--},\overline{F}_2(X_2)$ in the same way as $\overline{F}_N(X_N)$. We decompose the original problem into N one-stage initial state optimization problems.

1. $\overline{F}_1(X_1) = \underset{D_1}{\overline{\Omega}}\,[\overline{S}_1(X_1)]$

.

. where $\overline{S}_1(X_1) = \{\overline{r}_1(X_1,D_1),\ \forall\, D_1\}$

.

n. $\overline{F}_n(X_n) = \underset{D_n}{\overline{\Omega}}\,[\overline{S}_n(X_n)]$

.

. where $\overline{S}_n(X_n) = \{\overline{g}(\overline{r}_n(X_n,D_n),\ \overline{F}_{n-1}(t_n(X_n,D_n))),\ \forall\, D_n\}$

.

N. $\overline{F}_N(X_N) = \underset{D_N}{\overline{\Omega}}\,[\overline{S}_N(X_N)]$

where $\overline{S}_N(X_N) = \{\overline{g}(\overline{r}_N(X_N,D_N),\ \overline{F}_{N-1}(t_N(X_N,D_N))),\ \forall\, D_N\}$

Stating the N problems more compactly, we have

$$\overline{F}_n(X_n) = \underset{D_n}{\overline{\Omega}}\,[\overline{S}_N(X_n)], \qquad n = 1,2,\text{---},N$$

$$\text{where } \overline{S}_n(X_n) = \{\overline{g}(\overline{r}_n(X_n,D_n),\ \overline{F}_{n-1}(t_n(X_n,D_n))),\ \forall\, D_n\}$$

$$= \{\overline{Q}_n(X_n,D_n),\ \forall\, D_n\}$$

$$\text{and } \overline{Q}_n(X_n,D_n) = \overline{r}_n(X_n,D_n) \qquad n = 1$$

$$= r_n(X_n,D_n) \circ \overline{F}_{n-1}(t_n(X_n,D_n)) \quad n = 2,3,\text{---},N$$

This equation represents the usual recursion equation of dynamic programming, but for our application the usual max or min is replaced by the operator $\overline{\Omega}$.

We are now in a position to prove the decomposition in the case of multiple objective function. Given the separability and monotonicity of $\bar{g}_i$ we must prove

$$\underset{D_N,---,D_1}{\bar{\Omega}} [\bar{T}_N(X_N)] = \bar{\Omega} [\bar{S}_N(X_N)]$$

where

$$\bar{T}_N(X_N) = \bar{g}_1(\bar{r}_N(X_N,D_N), \bar{g}_2(\bar{r}_{N-1}(X_{N-1},D_{N-1}),--\bar{r}_1(X_1,D_1))), \quad \forall D_N,---D_1\}$$

$$\bar{S}_N(X_N) = \{\bar{g}_1(\bar{r}_N(X_N,D_N), \bar{F}_{N-1}(t_N(X_N,D_N))), \forall D_N\}.$$

For the convenience of proving the decomposition, we transform the multiple aggregate function,

$$\bar{R}(X_N,X_{N-1},---X_1, D_N,---D_1) \text{ to } \bar{R}(X_N,D_N,---,D_1)$$

by using the stage transformation

$$X_{n-1} = t_n(X_n,D_n).$$

The stage multiple return is

$$\bar{r}_n = \bar{r}_n(X_n,D_n).$$

From the transformation, it follows that X_n depends only on the decisions made prior to stage n (D_{n+1},---D_N) and X_N, that is,

$$\begin{aligned} X_n &= t_{n+1}(X_{n+1},D_{n+1}) = t_{n+1}(t_{n+2}(X_{n+2},D_{n+2}), D_{n+1}) \\ &= t_{n+1}(X_{n+2},D_{n+2},D_{n+1}) = t_{n+1}(t_{n+3}(X_{n+3},D_{n+3}), D_{n+2}, D_{n+1}) \\ &= --- = t_{n+1}(X_N,D_N,---,D_{n+1}). \end{aligned}$$

It then follows, by combining the above equation with the return function, that the return from stage n depends *only on the decisions* $(D_n, D_{n+1}, ---, D_N)$ and X_N, that is,

$$\bar{r}_n = \bar{r}_n(X_n, D_n) = \bar{r}_n(t_{n+1}(X_N, D_N, ---, D_{n+1}), D_n)$$

$$= \bar{r}_n(X_N, D_N, ---D_n).$$

The multiple aggregate return function $\bar{R}$ from stages one through N is some function of the individual stage returns written as

$$\bar{R}(X_N, X_{N-1}, ---X_1, D_N, ---, D_1) = \bar{g}(\bar{r}_N(X_N, D_N),$$

$$\bar{r}_{N-1}(X_{N-1}, D_{N-1}), ---, \bar{r}_1(X_1, D_1)).$$

However, as just explained, $(X_{N-1}, ---, X_1)$ can be eliminated from the individual stage returns and consequently from the total return. Thus an alternate expression for $\bar{R}$ is

$$\bar{R}(X_N, D_N, ---, D_1) = \bar{g}(\bar{r}_N(X_N, D_N), \bar{r}_{N-1}(X_N, D_N, D_{N-1}), ---,$$

$$\bar{r}_1(X_N, D_N, ---, D_1))$$

Let the multiple aggregate return function be separable as follows

$$\bar{R}(X_N, D_N, ---, D_1) = \bar{g}_1(\bar{r}_N(X_N, D_N), \bar{g}_2(\bar{r}_{N-1}(X_N, D_N, D_{N-1}), ---,$$

$$r_1(X_N, D_N, ---, D_1)))$$

and g_1 be monotonic nondecreasing function of g_2 $\forall$ fixed r_N

where

$$\overline{g}_i = \begin{bmatrix} g_i^1 \\ g_i^2 \\ \cdot \\ \cdot \\ \cdot \\ g_i^L \end{bmatrix} \qquad i = 1,2$$

Let

$$\overline{F}_N(X_N) = \underset{D_N,---,D_1}{\overline{\Omega}} [\overline{T}_N(X_N)]$$

where

$$\overline{T}_N(X_N) = \{\overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{g}_2(\overline{r}_{N-1}(X_N,D_N,D_{N-1}),---,$$

$$\overline{r}_1(X_N,D_N,D_{N-1},---,D_1))),\ \forall\ D_N,---,D_1\}$$

$\overline{T}_N(X_N)$ is the set of all feasible values of

$$\overline{g}_1(\overline{r}_N(X_N,D_N),\ \overline{g}_2(\overline{r}_{N-1}(X_{N-1},D_N,D_{N-1}),---,$$

$$\overline{r}_1(X_N,D_N,---,D_1)))$$

for all $D_N,D_{N-1},---,D_1$

and $\underset{D_N,---,D_1}{\overline{\Omega}} [\overline{T}_N(X_N)]$ is the nondominated solutions among the values in the set of $\overline{T}_N(X_N)$.

Let $\overline{F}'_N(X_N) = \underset{D_N}{\overline{\Omega}} [S_N(X_N)]$

where

$$\overline{S}_N(X_N) = \overline{g}_1(\overline{r}_N(X_N,D_N),\ \underset{D_{N-1},---,D_1}{\overline{\Omega}} [\overline{T}_{N-1}(X_{N-1})]),\ \forall\ D_N\}$$

and $\overline{T}_{N-1}(X_{N-1}) = \{\overline{g}_2(\overline{r}_{N-1}(X_N,D_N,D_{N-1}),---,\ \overline{r}_1(X_N,D_N,---,D_1)),$

$$\forall\ D_{N-1},---,D_1\}$$

we must prove

$$F_N(X_N) = F_N'(X_N).$$

For notational simplification let

$$\{w\} = \overline{F}_N'(X_N)$$

$$\{z\} = \overline{F}_N(X_N).$$

Note a $z \in \overline{F}_N(X_N)$ is nondominated in comparison with all other feasible values, while a $w \in \overline{F}_N'(X_N)$ is nondominated in comparison with the values enumerated in the evaluation of $\underset{D_N}{\overline{\Omega}}\ [\overline{S}_N(X_N)]$.

4. <u>Theorem [2]</u>

$\{w\} \subseteq \{z\}$

PROOF

Suppose $\exists$ w such that $w \in \{w\}$

then from $w \in \{w\} = \underset{D_N}{\Omega}\ [\overline{S}_N(X_N)]$

for some $D_N{}^*,D_{N-1}{}^*,---,D_1{}^*$

$$w = \overline{g}_1[\overline{r}_N(X_N,D_N{}^*),\ \overline{g}_2(\overline{r}_{N-1}(X_N,D_N{}^*,D_{N-1}{}^*),---,$$

$$\overline{r}_1(X_N,D_N{}^*,---,D_1{}^*))].$$

Then it is false that there exist $\tilde{D}_N,\tilde{D}_{N-1},---,\tilde{D}_1$ such that

$$\overline{g}_1(\overline{r}_N(X_N,\tilde{D}_N),\ \overline{g}_2(\overline{r}_{N-1}(X_N,\tilde{D}_N,\tilde{D}_{N-1}),---,\overline{r}_1(X_N,\tilde{D}_N,\tilde{D}_{N-1},---,\tilde{D}_1))) \geq w$$

for any $\tilde{D}_{N-1},---,\tilde{D}_1$ since w is nondominated.

But if $w \not\in \{z\}$, this cannot be for either of two reasons:

(1) either w is dominated in $\{z\}$, then there exist $\tilde{D}_N,\tilde{D}_{N-1},---,\tilde{D}_1$ such that

$$\bar{g}_1(\bar{r}_N(X_N,\tilde{D}_N),\ \bar{g}_2(\bar{r}_{N-1}(X_N,\tilde{D}_N,\tilde{D}_{N-1}),---,\bar{r}_1(X_N,\tilde{D}_N,---,\tilde{D}_1))) \geq w.$$

This contradicts the assumption $w \in \{w\}$.

(2) or else $w \not\in \{z\}$ can occur because w was not enumerated when $\{z\}$ was computed,

but this cannot be because

$$\{w\} \subseteq \{\bar{g}_1(\bar{r}_N(X_N,D_N),\ \bar{g}_2(\bar{r}_{N-1}(X_N,D_N,D_{N-1}),---,\bar{r}_1(X_N,D_N,---,D_1))),\ \forall\, D_N,---,D_1\}$$

that is,

$$\{w\} \subseteq \{\bar{T}_N(X_N)\}.$$

This can be proved by the following proof

$$\underset{D_{N-1},---,D_1}{\bar{\Omega}} [\bar{T}_{N-1}(X_{N-1})] \subseteq [\bar{T}_{N-1}(X_{N-1})]$$

We add $\bar{r}_N(X_N,D_N)$ to both sides, then

$$\{\bar{g}_1(\bar{r}_N(X_N,D_N),\ \underset{D_{N-1},---,D_1}{\Omega} [\bar{T}_{N-1}(X_{N-1})])\} \subseteq \{\bar{g}_1(\bar{r}_N(X_N,D_N),\ \bar{T}_{N-1}(X_{N-1}))\}.$$

By <u>monotonicity</u> of $\bar{g}_1$, this equation is true.

And this can be expressed as follows using the definition of $\bar{S}_N(X_N)$, $\bar{T}_N(X_N)$

$$\{\bar{S}_N(X_N)\} \subseteq \{T_N(X_N)\}$$

and
$$\underset{D_N}{\bar{\Omega}}\ [\bar{S}_N(X_N)] \subseteq \{\bar{S}_N(X_N)\}$$

thus
$$\underset{D_N}{\bar{\Omega}}\ (\bar{S}_N(X_N)) \subseteq \{\bar{S}_N(X_N)\} \subseteq \{\bar{T}_N(X_N)\}$$

$$\{w\} \subseteq \{T_N(X_N)\}.$$

So w was enumerated when $\{z\}$ was computed. Thus if $W \varepsilon \{w\}$, we also have $w \varepsilon \{z\}$.

5. Theorem [3]

$\{z\} \subseteq \{w\}$

PROOF

Suppose there exist z such that $z \varepsilon \{z\}$

(1) Then from

$$z \varepsilon \{z\} = \underset{D_N,---,D_1}{\bar{\Omega}}\ [\bar{T}_N(X_N)]$$

where

$$\bar{T}_N(X_N) = \bar{g}_1(\bar{r}_N(X_N,D_N),\ \bar{g}_2(\bar{r}_{N-1}(X_N,D_N,D_{N-1}),---,$$
$$\bar{r}_1(X_N,D_N,---,D_1))),\ \forall\ D_{N-1},--D_1\}$$

for some $D_N^*,---D_1^*$

$$z = \bar{g}_1(\bar{r}_N(X_N,D_N^*),\ \bar{g}_2(\bar{r}_{N-1}(X_N,D_N^*,D_{N-1}^*),---,\bar{r}_1(X_N,D_N^*,---,D_1^*)))$$

and for all $D_N,D_{N-1},---,D_1$.

The following equality is false:

$$\bar{g}_1(\bar{r}_N(X_N,D_N),\ \bar{g}_2(\bar{r}_{N-1}(X_N,D_N,D_{N-1}),---,$$

$$\bar{r}_1(X_N,D_N,---,D_1))) \geq z$$

(2) Also if we fix X_N, D_N^*, then

$$\bar{g}_2(\bar{r}_{N-1}(X_N,D_N^*,D_{N-1}^*),---,\bar{r}_1(X_N,D_N^*,---D_1^*))$$

$$\varepsilon \quad \overline{\Omega}_{D_{N-1},---,D_1} \quad [\bar{T}_{N-1}(X_{N-1})]$$

where

$$\bar{T}_{N-1}(X_{N-1}) = \{\bar{g}_2(\bar{r}_{N-1}(X_N,D_N,D_{N-1}),---,\bar{r}_1(X_N,D_N,---,D_1),$$

$$\forall D_{N-1},---,D_1\}.$$

Since otherwise its domination (say by,

$$\bar{g}_2(\bar{r}_{N-1}(X_N,D_N^*,\tilde{D}_{N-1}),\ \bar{r}_{N-2}(X_N,D_N^*,---,\tilde{D}_{N-2}),---,$$

$$\bar{r}_1(X_N,D_N^*,---,\tilde{D}_1))))$$

would imply domination of z by

$$\bar{g}_1(\bar{r}_N(X_N,D_N^*),\ \bar{g}_2(\bar{r}_{N-1}(X_N,D_N^*,\tilde{D}_{N-1}),---,\bar{r}_1(X_N,D_N^*,---,\tilde{D}_1)))$$

(by monotonicity of $\bar{g}_1$) implying $z \not\in \{z\}$.

And now suppose $z \not\in \{w\}$, this means either

(a) z was not enumerated when $\{w\}$ was computed. But this cannot be by (2), or

(b) z was dominated in $\{w\}$, then there is some $\tilde{D}_N,---,\tilde{D}_1$ combination such that

$$\bar{g}_1(\bar{r}_N(X_N,\tilde{D}_N),\ \bar{g}_2(\bar{r}_{N-1}(X_N,\tilde{D}_N,\tilde{D}_{N-1}),---,$$

$$\bar{r}_1(X_N,\tilde{D}_N,\tilde{D}_{N-1},---,D_1))) \geq z,$$

but this contradicts (1).

Thus if $z \in \{z\}$, we also have $z \in \{w\}$.

6. Theorem [4]

$\{w\} = \{z\}$

PROOF

By Theorems [2] and [3].

From Theorems [2], [3] [4], we have shown that if $\bar{g}_1$ is a monotonically nondecreasing function of $\bar{g}_2$ for every $\bar{r}_N$ then

$$\underset{D_N,---,D_1}{\bar{\Omega}} [\bar{T}_N(X_N)] = \underset{D_N}{\bar{\Omega}} [\bar{S}_N(X_N)]$$

where

$$\bar{T}_N(X_N) = \{\bar{g}_1(\bar{r}_N(X_N,D_N),\ \bar{r}_{N-1}(X_{N-1},D_{N-1}),---,\bar{r}_1(X_1,D_1)),$$

$$\forall\ D_N,---,D_1\}$$

$$\bar{S}_N(X_N) = \{\bar{g}_1(\bar{r}_N(X_N,D_N),\ \underset{D_{N-1},---,D_1}{\bar{\Omega}} [\bar{T}_{N-1}(X_{N-1})]),\ \forall\ D_N\}.$$

Thus, the stage by stage dynamic programming recursion will generate exactly the set of all nondominated solutions to the original multiple criterion optimization problem.

III. BASIC COMPUTATION

A. GENERAL SCHEME (DISCRETE CASE)

Computational aspects of dynamic programming approach to MCO concern the solution of the multiple recursion equation. In the last chapter we defined the multiple recursion equation as follows.

$$\bar{F}_n(X_n) = \underset{D_n}{\bar{\Omega}} [\bar{S}_n(X_n)] \tag{1}$$

where

$$\bar{S}_n(X_n) = \{\bar{Q}_n(X_n,D_n),\ \forall D_n\}$$

$$\bar{Q}_n(X_n,D_n) = \bar{r}_n(X_n,D_n) \qquad n = 1 \tag{2}$$

$$= \bar{r}_n(X_n,D_n) \circ \bar{F}_{n-1}(t_n(X_n,D_n)) \tag{3}$$

$$n = 2,\ldots,N$$

with $\qquad X_{n-1} = t_n(X_n,D_n).$

First, we start with $n = 1$ (Stage 1) calculating

$$\bar{r}_1(X_1,D_1) = \begin{bmatrix} r_1^1(X_1,D_1) \\ r_1^2(X_1,D_1) \\ \vdots \\ r_1^L(X_1,D_1) \end{bmatrix}$$

since $n = 1$. From Equation (2),

$$\bar{Q}_1(X_1,D_1) = \bar{r}_1(X_1,D_1).$$

Equation (1) is then used to obtain $\overline{F}_1(X_1)$ and $D_1^*(X_1)$ by using the nondominated method. These are saved for future calculations. But $\overline{Q}_1$ and $\overline{r}_1$ are no longer needed. Since $n \neq N$, we increase n by 1 and calculate

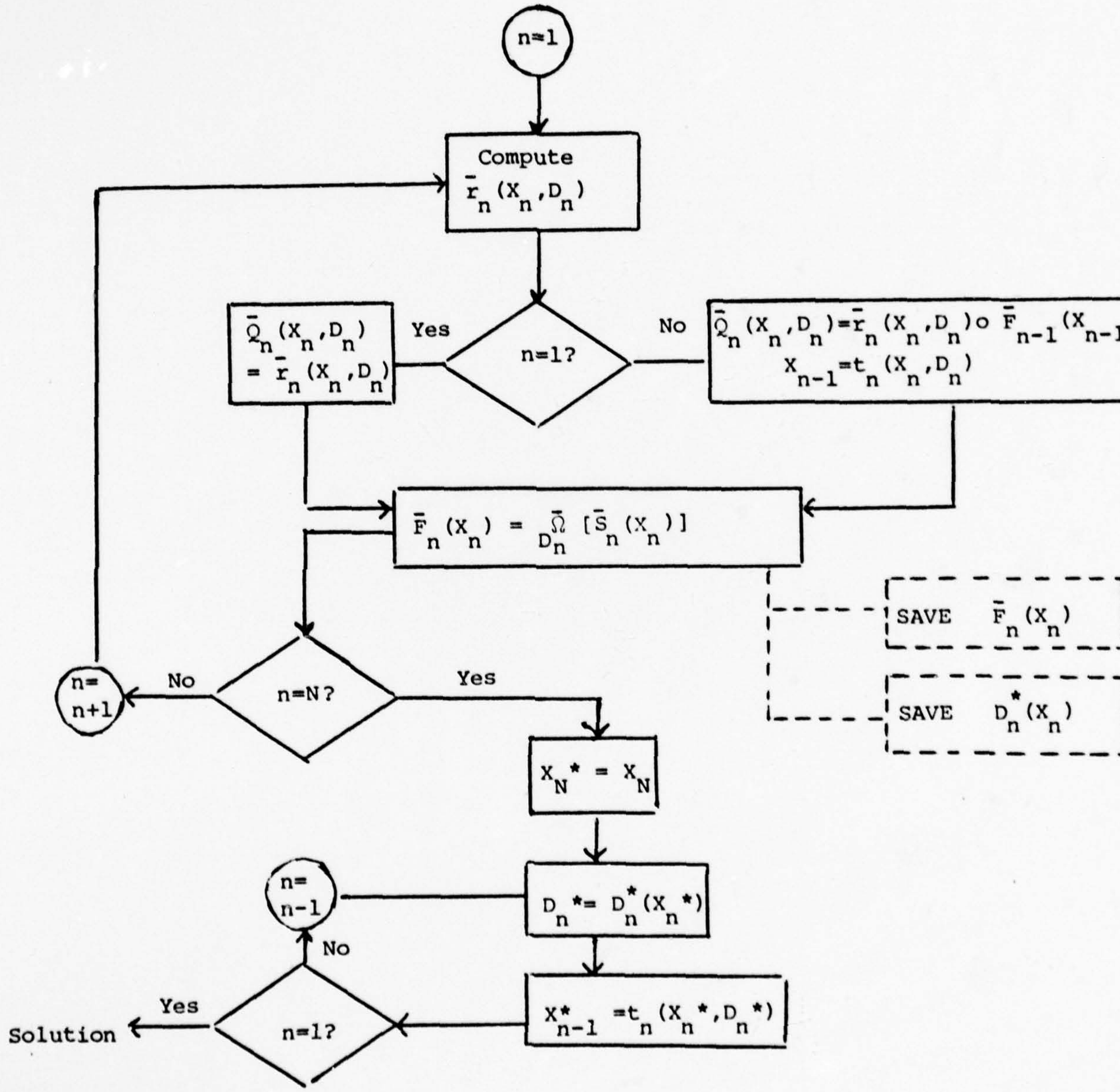
$$\overline{r}_2(X_2,D_2)$$

for n = 2. $\overline{Q}_2$ is calculated from Equation (3) by appropriately combining $\overline{r}_2$ and $\overline{F}_1$. $\overline{F}_1$ has now served its purpose and may be discarded.

The calculations continue similarly for $n = 3,\ldots,N$. Thus we obtain $\overline{F}_N(X_N)$, the nondominated solutions from the N-stage system, and $D_n^*(X_n)$, $(n = 1,\ldots,N)$, the optimal nth-stage decision function of the nth-stage inputs.

The next step is to determine the optimal inputs X_n^* $(n = 1,\ldots,N-1)$ and the optimal decision D_n^* $(n = 1,\ldots,N)$. We have reached the point in the *flow chart* (see Fig. 1) where, for the first time, the answer to the question "Does n = N?" is "yes."

We assume there is a prescribed value of X_N denoted by X_N^*, D_N^* is obtained immediately from the stored function $D_N(X_N)$. Then X_{N-1}^* is calculated from $X^*_{N-1} = t_N(X_N^*,D_N^*)$. Since $n \neq 1$, we compute D^*_{N-1} from X^*_{N-1} and the stored function $D_{N-1}^*(X_{N-1})$. When the answer to the question "Does n = 1?" is "yes," the optimal solution is obtained.



[Figure 1]. Flowchart of General Computation

B. TABULAR COMPUTATION

The term tabular computations is used to describe dynamic programming analyses in which the optimal decision functions and optimal return functions are given in discrete form in lists or tables. In other words, for each feasible value of X_n, n=1,...,N there is an entry in a table containing values of $D_n^*(X_n)$ and $\overline{F}_n(X_n)$. (See Table 1.)

Naturally, it is only possible to give this information in a table when there are a finite number of feasible values for X_n. But tabular representation of continuous data is possible if we make discrete approximations. For example, suppose the feasible values of X_n are the integer numbers from 1 to K_n. For $X_n = K$ $(=1,\ldots,K_n)$. $D_n^*(X_n=K)$ is the set of all D_n which are nondominated points of

$$\overline{Q}_n(X_n=K,D_n) = \overline{r}_n(X_n=K,D_n) \circ \overline{F}_{n-1}(t_n(X_n=K,D_n))$$

and $\overline{F}_n(X_n=K)$ is the nondominated solution value of the above expression. At the first stage, the feasible values of X_1 are the integer numbers from 1 to K_n. Suppose that for each feasible value of $X_1(0,1,2,\ldots,K_n)$, the feasible decision variable value is $D_1 = 0,1,\ldots,X_1$. (See Table 1.)

For each value of $X_1, r_{.}(X_1,D_1)$ is calculated when X_1=K. The multiple return functions are

$$\begin{bmatrix} r_1^1(K,0), & r_1^2(K,0), --- , & r_1^L(K,0) \\ r_1^1(K,1), & r_1^2(K,1), --- , & r_1^L(K,1) \\ \vdots & & \\ r_1^1(K,K), & r_1^2(K,K), --- , & r_1^L(K,K) \end{bmatrix}$$

TABLE 1

X_n	Feasible D_n	$\bar{r}_n(X_n,D_n)$	$\bar{Q}_n(X_n,D_n)$	$\bar{F}_n(X_n)$
0	0	$\bar{r}_n(0,0)$	$\bar{r}_n(0,0) \circ \bar{F}_{n-1}(t_n(0,0))$	$\underset{D_n}{\bar{\Omega}} [\bar{S}_n(0)]$ $\bar{S}_n(0)=\{\bar{Q}_n(0,D_n), \forall D_n\}$
1	0	$\bar{r}_n(1,0)$	$\bar{r}_n(1,0) \circ \bar{F}_{n-1}(t_n(1,0))$	$\underset{D_n}{\bar{\Omega}} [\bar{S}_n(1)]$
	1	$\bar{r}_n(1,1)$	$\bar{r}_n(1,1) \circ \bar{F}_{n-1}(t_n(1,1))$	$\bar{S}_n(1)=\{\bar{Q}_n(1,D_n), \forall D_n\}$
2	0	$\bar{r}_n(2,0)$	$\bar{r}_n(2,0) \circ \bar{F}_{n-1}(t_n(2,0))$	$\underset{D_n}{\bar{\Omega}} [\bar{S}_n(2)]$
	1	$\bar{r}_n(2,1)$	$\bar{r}_n(2,1) \circ \bar{F}_{n-1}(t_n(2,1))$	$\bar{S}_n(2)=\{\bar{Q}_n(2,D_n), \forall D_n\}$
	2	$\bar{r}_n(2,2)$	$\bar{r}_n(2,2) \circ \bar{F}_{n-1}(t_n(2,2))$	
K_n	0	$\bar{r}_n(K_n,0)$	$\bar{r}_n(K_n,0) \circ \bar{F}_{n-1}(t_n(K_n,0))$	$\underset{D_n}{\bar{\Omega}} [\bar{S}_n(3)]$
	1	$\bar{r}_n(K_n,1)$	$\bar{r}_n(K_n,1) \circ \bar{F}_{n-1}(t_n(K_n,1))$	$\bar{S}_n(3)=\{\bar{Q}_n(3,D_n), \forall D_n\}$
	2	$\bar{r}_n(K_n,2)$	$\bar{r}_n(K_n,2) \circ \bar{F}_{n-1}(t_n(K_n,2))$	
	$\vdots$	$\vdots$	$\vdots$	
	K_n	$\bar{r}_n(K_n,K_n)$	$\bar{r}_n(K_n,K_n) \circ \bar{F}_{n-1}(t_n(K_n,K_n))$	

and D_1 is a nondominated point if there does not exist another D_1' such that

$$r_1^i(K,D_1') \geqq r_1^i(K,D_1) \qquad \forall_1$$

$$r_1^i(K,D_1') \neq r_1^i(K,D_1) \qquad \text{At least one } i$$

$$D_1 = 0,1,2,\ldots,K \qquad D_1 \neq D_1'$$

$$D_1' = 0,1,2,\ldots,K$$

that is

$$\overline{F}_1(X_1) = \underset{D_1}{\overline{\Omega}} [\overline{S}_1(X_1)]$$

where

$$\overline{S}_1(X_1) = \{\overline{Q}_1(X_1,D_1) \ \forall \ D_1\}$$

$$\overline{Q}_1(X_1D_1) = \overline{r}_1(X_1D_1).$$

The tables of $\overline{F}_1(X_1)$ and $D_1^*(X_1)$ are saved for future calculations. (See Table 2.)

TABLE 2

(SUMMARY TABLE)

List of Feasible Values for State Variable X_n	Associated Nondominated Decision Variable $D_n^*(X_n)$	Associated Nondominated Solution $\bar{F}_n(X_n)$
$X_n = 0$	$D_n^*(X_n = 0)$	$\bar{F}_n(X_n = 0)$
.	.	.
.	.	.
.	.	.
$X_n = K$	$D_n^*(X_n = K)$	$\bar{F}_n(X_n = K)$
.	.	.
.	.	.
.	.	.
$X_n = K_n$	$D_n^*(X_n = K_n)$	$\bar{F}_n(X_n = K_n)$

Finally, suppose that at the last stage (N), the feasible value of the state variable is just one (K_N) and the feasible values of decision variable are 0 through K_N, and every entry in the table is the same as in Table 1. The calculations are basically done in two parts; first, the calculation of $F_N(X_N)$ and $D_1^*(X_1)$, $D_2^*(X_2)$,---, $D_N^*(X_N)$ and the second, the tracing of the nondominated policy D_N^*,---,D_1^*.

At the last stage we get $X_N{}^* = K_n$ and $D_N{}^*$ then by transformation $X_{N-1} = t_N(X_N{}^*, D_N{}^*)$ we can compute X_{N-1}^* and by the summary table of Stage N-1 (we saved each $\overline{F}_{N-1}(X_{N-1})$, X=1,---,$K_N$,$D_{N-1}^*(X_{N-1})$). We can compute $D_{N-1}^*(X_{N-1})$. We will repeat this until n=1; then at least we will get D_N^*, D_{N-1}^*,---,$D_1{}^*$

C. AN EXAMPLE: ONE STATE VARIABLE AND IRREGULAR RETURNS AND TWO OBJECTIVE FUNCTION

This example shows how tabular computations are organized and provides some clues to the number of computations involved when the recursion equations are solved by enumeration.

Example Problem.

Five OA students meet to work the homework problems in two courses, D.P. and Stochastic Models; both have three problems. They agree to split into three teams...each team assigned one D.P. problem and one stochastic problem. The probability of success on a problem depends on the number of students working on it. (No student can work on more than one team. Assume all students' capabilities are equal.)

D.P.

Number of Students	Probability		
	Problem 1	Problem 2	Problem 3
0	.0	.0	.0
1	.1	.6	.3
2	.2	.8	.5
3	.3	.9	.6
4	.4	.9	.7
5	.5	.9	.7

Stochastic Project

Number of students	Probability		
	Problem 1	Problem 2	Problem 3
0	.0	.0	.0
1	.7	.2	.1
2	.8	.3	.4
3	.9	.4	.5
4	.9	.5	.6
5	.9	.6	.7

How many students should be on each team to maximize the joint probability that all problems in D.P. are solved and the joint probability that all in Stochastic are solved? (At least one student must be on each team.) The formulation of problem.

$$\text{Maximize} \quad \prod_{n=1}^{3} P_n^{\,1}(D_n)$$

$$\prod_{n=1}^{3} P_n^{\,2}(D_n)$$

$$\text{Subject to} \quad \sum_{n=1}^{3} D_n \leq 5$$

$$D_n \geq 0, \; D_n = \text{integer}$$

where

P^1 = D P probability

P^2 = stochastic probability

n = 1,2,3(number of problems)

Multiple recursion equations

$$\overline{F}_1(X_1) = \underset{D_1}{\Omega} [\overline{S}_1(X_1)], \text{ where } \overline{S}_1(X_1) = \{\overline{P}_1(X_1,D_1), \forall D_1\}$$

$$\overline{F}_2(X_2) = \underset{D_2}{\Omega} [\overline{S}_2(X_2)], \text{ where } \overline{S}_2(X_2) = \{\overline{Q}_2(X_2,D_2), \forall D_2\}$$

$$\overline{Q}_2(X_2) = \overline{P}_2(X_2,D_2) \circ \overline{F}_1(X_2-D_2)$$

$$\overline{F}_3(X_3) = \underset{D_3}{\Omega} [\overline{S}_3(X_3)], \text{ where } \overline{S}_3(X_3) = \{\overline{Q}_3(X_3,D_3), \forall D_3\}$$

$$\overline{Q}_3(X_3) = \overline{P}_3(X_2,D_2) \circ \overline{F}_2(X_3-D_3)$$

$$\overline{P} = [P^1, P^2]^T$$

Computations

Stage 1

X_1	D_1	$P_1^1(D_1)$	$P_1^2(D_1)$	$\overline{F}_1(X_1)$	
1	1	.1	.7	.1	.7
2	1	.1	.7		
	2	.2	.8	.2	.8
3	1	.1	.7		
	2	.2	.8		
	3	.3	.9	.3	.9

Summary Table 1

X_1	$\bar{F}_1(X_1)$		$D_1^*(X_1)$
1	.1	.7	1
2	.2	.8	2
3	.3	.9	3

Stage 2

X_2	D_2	$t_2(X_2,D_2)$	$P_2^1(D_2)$	$P_2^2(D_2)$	$\bar{F}_1(t_2(X_2,D_2))$		$\bar{Q}_2(X_2,D_2)$		$\bar{F}_2(X_2)$	
2	1	1	.6	.2	.1	.7	.06	.14	.06	.14
3	1	2	.6	.2	.2	.8	.12	.16	.12	.16
	2	1	.8	.3	.1	.7	.08	.21	.08	.21
4	1	3	.6	.2	.3	.9	.18	.18	.18	.18
	2	2	.8	.3	.2	.8	.16	.24	.16	.24
	3	1	.9	.4	.1	.7	.09	.28	.09	.28

Summary Table 2

X_2	$\bar{F}_2(X_2)$		$D_2^*(X_2)$
2	.06	.14	1
3	.12	.16	1
	.08	.21	2
4	.18	.18	1
	.16	.24	2
	.09	.28	3

Stage 3

X_3	D_3	$t_3(X_3,D_3)$	$P_3^1(D_3)$	$P_3^2(D_3)$	$\bar{F}_2(t_3(X_3,D_3))$		$\bar{Q}_3(X_3,D_3)$		$\bar{F}_3(X_3)$	
5	1	4	.3	.1	.18	.18	0.054	0.018		
					.16	.24	0.048	0.024		
					.09	.28	0.027	0.028		
	2	3	.5	.4	.12	.16	0.060	0.064	0.060	0.064
					.08	.21	0.040	0.084	0.040	0.084
	3	2	.6	.5	.6	.14	0.036	0.070		

By back tracking we find the nondominated solution to contain two parts imply:

(1) $D_1 = 2$ D.P. = 0.06

$D_2 = 1$ Stochastic = 0.064

$D_3 = 2$

(2) $D_1 = 1$ D.P. = 0.04

$D_2 = 2$ Stochastic = 0.084

$D_3 = 2$

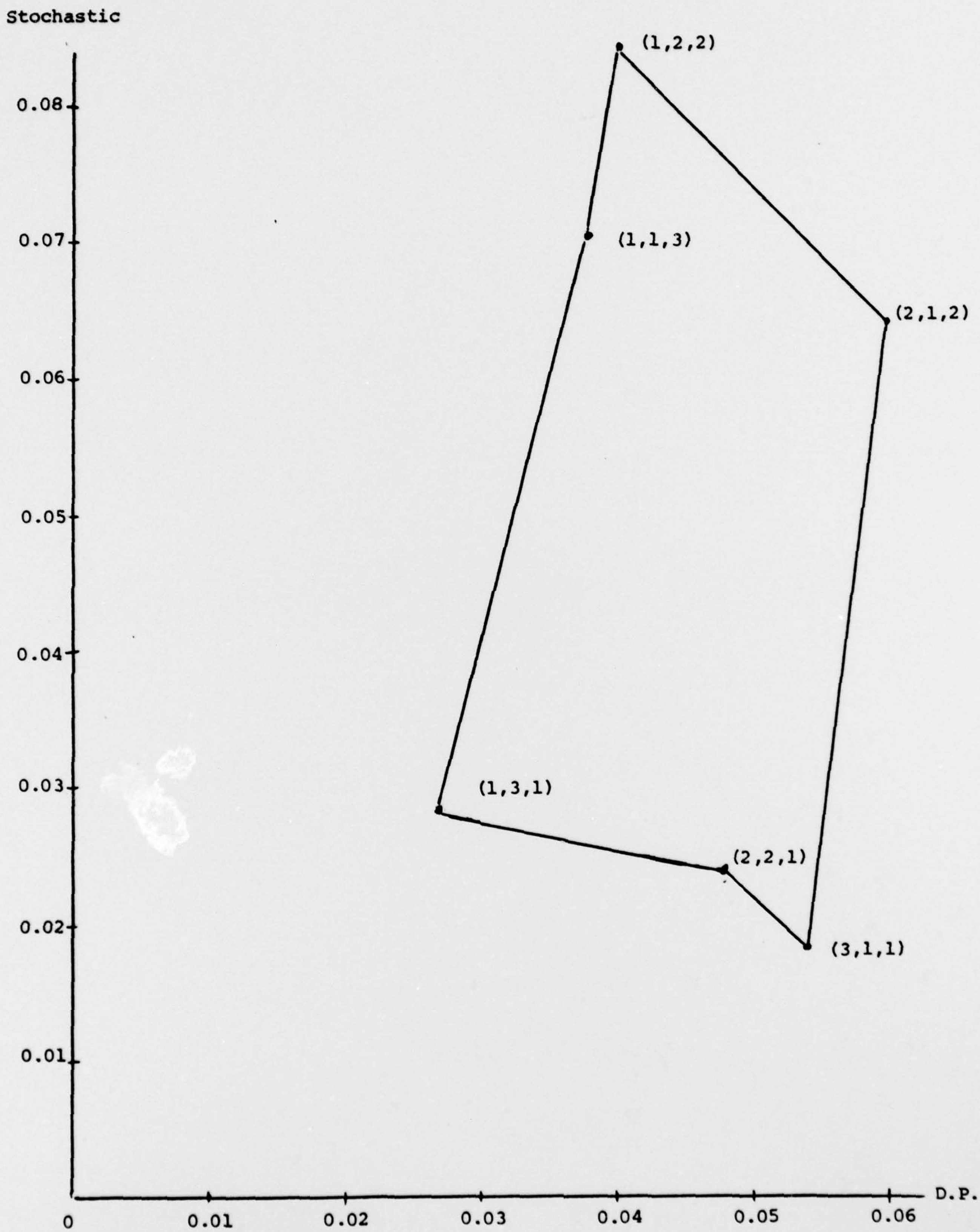


Figure 2. Nondominated solutions in two-dimensional criteria space.

D. THE CONTINUOUS VARIABLE CASE

Let us consider the problems involved in solving MCO problem when we no longer require that the D_j be integer. We shall then also drop the restriction that a_j , K_n be integers. That is

$$\bar{F}_N(X_N) = \underset{D_1,---,D_N}{\bar{\Omega}} [\bar{T}_N(X_N)],$$

where

$$\bar{T}_N(X_N) = \{\bar{g}(\bar{r}_N(X_N,D_N),\ \bar{r}_{N-1}(X_{N-1},D_{N-1}),---,\bar{r}_1(X_1,D_1)),$$

$$\forall\ D_N,---,D_1\}$$

In carrying out the optimization we consider only non-negative D_j which satisfy

$$\sum_{j=1}^{n} a_j D_j \leq K_n.$$

Precisely as before,

$$\bar{F}_1(X_1) = \underset{o\leq D_1\leq K_n/a_1}{\bar{\Omega}} [\bar{S}_1(X_1)]$$

where

$$\bar{S}_1(X_1) = \{r_1(X_1,D_1),\ \forall\ D_1\}.$$

Note that D_1 varies up to K_n/a_1 not $[K_n/a_1]$. The same recurrence relations follow that applied to the case where the D_j were to be integer. That is

$$\bar{F}_n(X_n) = \underset{o\leq D_n\leq K_n/a_n}{\bar{\Omega}} [\bar{S}_n(X_n)]$$

where

$$\bar{S}_n(X_n) = \{\bar{Q}_n(X_n,D_n), \forall D_n\},$$

$$\bar{Q}_n(X_n,D_n) = \bar{r}_n(X_n,D_n) \circ \bar{F}_{n-1}(X_n-a_nD_n).$$

In order to apply the dynamic programming computation using the tabular form, the decision variables must be discrete. So first we must decide on a discrete grid to approximate the continuous variables. If the grid size is Δ, the length of interval (L) then the number of points (p) on which the function is defined is

$$p = \frac{L}{\Delta} + 1.$$

Thus for the fixed L, p is inversely proportional to Δ. If p is too large, then the D.P. is almost impossible to calculate, we must reduce the size by making the grid coarser. That is, by changing the grid from Δ to a larger value δ there is a corresponding decrease in p.

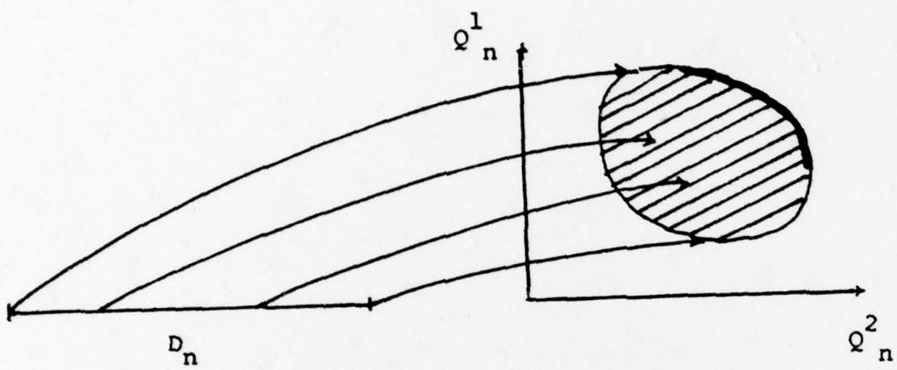
To see how the computations decrease as the grid increases, suppose we want the optimization of $\bar{Q}_n(X_n,D_n)$ for fixed X_n, and $0 \leq D_n \leq 20{,}000$ on the grid $d_n = 0.1,---,20{,}000$. If we start initial grid, there are 20,001 points at which the function is to be evaluated.

By increasing the initial grid size to d = 0, 100, 200,---,20,000 and evaluating the function at 201 points, the interval of optimality is reduced to a length of 200. Then another 201 evaluations of the function will yield the integer solution. But the number of computations can be reduced still further by starting with a very coarse grid d = 0, 1000, 2000,---, 20,000. Then the interval of optimality is reduced to a length of 20. Then another 21 evaluations of the function will yield the integer solution.

Assume that the function $\overline{F}_{n-1}(X_{n-1})$ is known, then $\overline{Q}_n$ is defined over a continuum of values D_n in the interval $o \le D_n \le K_n/a_n$.

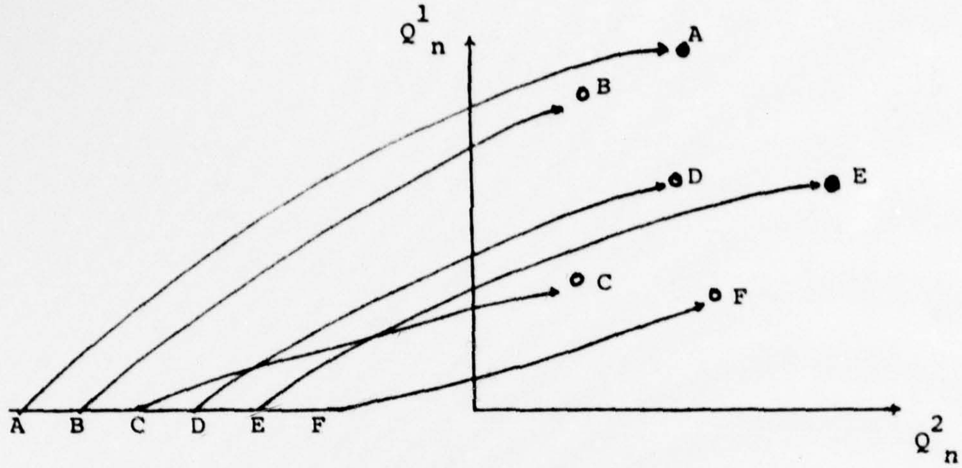
$$\text{If } \overline{Q}_n = \begin{matrix} Q^1_n \\ \\ Q^2_n \end{matrix} \quad \text{is plotted (as } Q^1_n(X_n,D_n) \text{ vs } Q^2_n(X_n,D_n) \text{)} \\ \text{(at fixed } X_n\text{)}$$

it might look like a space or a curve. (See Figure 1.)



[Figure 1]

And the nondominated solution set is the thick line. We would like to determine the set of values $\hat{D}_n(X_n)$ at which $\overline{Q}_n$ assumes its nondominated solution. To reduce as much as possible the computational effort required, one might do a rough initial search using a coarse grid of D_n, $(D_n = 0, \delta, 2\delta, \text{---} D_n)$. (See Figure 2.)



[Figure 2]

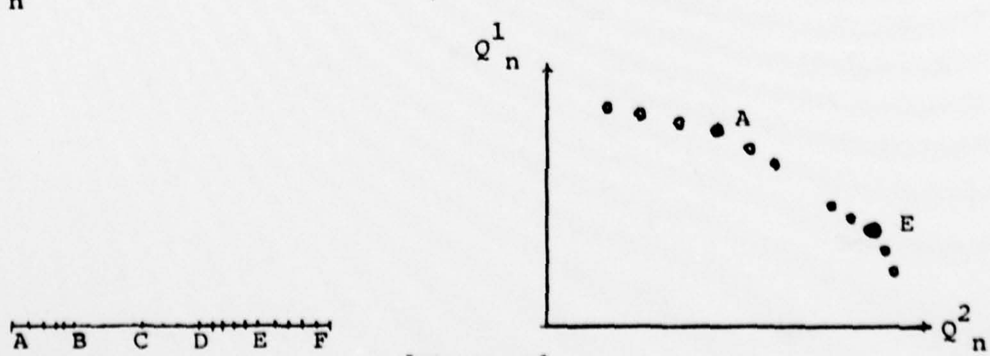
From this coarse grid of decision variables an approximate nondominated policy would be found from Figure 2, points A and E. Then we can conclude that

$$d_\delta^* - \delta \le d\Delta^* n \le d\delta^* + \delta$$

that is, $\quad 0 \le d\Delta^* n \le \delta$

$$3\delta \le d\Delta^* n \le 5\delta.$$

A finer grid would be used in the neighborhood of the nondominated point to determine more nondominated values or to locate them more accurately. The procedure of using a finer grid might be repeated several times to obtain the desired number of nondominated points in $D_n(X_n)$.



[Figure 3]

Consider next the problems which arise in the tabulation of the $\overline{F}_n(X_n)$. A difficulty is encountered here because $\overline{F}_n(X_n)$ is defined for all X_n, $o \le X_n \le K_n$. Furthermore, examination of the search procedure introduced above shows that $\overline{F}_n(X_n)$ may actually be needed for any X_n in this interval, due to the fact that when finer and finer grids are used to determine $\overline{F}_n(X_n)$ for a given S_n, $\overline{F}_{n-1}(X_{n-1})$ may be required for a very large number of different arguments. It would usually be impossible to tabulate the $\overline{F}_n(X_n)$ for all possible X_n that might be required with the search procedure described above. Clearly, if this cannot be done, we must tabulate the $\overline{F}_n(X_n)$ for a smaller number of X_n.

In dynamic programming the main benefit of a coarse grid on the decision variable is that it produces a corresponding coarse grid on state variables.

In the relation

$$X_{n-1} = t_n(X_n, D_n)$$

the number of feasible values of X_{n-1} is roughly proportional to the number of feasible values of D_n. By reducing the number of feasible values of X_{n-1}, we evaluate $\overline{F}_{n-1}(X_{n-1})$ fewer times and the computational procedure is sames as discrete case. The successive grids used in a dynamic programming analysis depend on the number of values tolerable for the state variables and the accuracy required.

E. COMPUTATIONAL EFFICIENCY OF THE METHOD

Let us now compare the computational efficiency of the dynamic programming approach to solving the MCO problem with that of simply enumerating all possible sets of non-negative integers D_j which satisfy

the constraint

$$\sum_{j=1}^{N} a_j D_j < Kn.$$

For illustrative purposes

let $a_j = 1$ then $\sum_{j=1}^{N} a_j D_j \leq K_n$ become

$$\sum_{j=1}^{N} D_j \leq Kn \tag{1}$$

and only one constraint and one variable.

1. Direct Enumeration

In the case of N stages and L objective functions and the constraint is (1) then the number of evaluations of objective functions is

$$L \sum_{K=0}^{Kn} \binom{n+K-1}{K}$$

and the number of comparison for choosing nondominated solutions is

$$\sum_{K=0}^{Kn} \frac{\left[\binom{n+K-1}{K}-1\right]\binom{n+K-1}{K}}{2} .$$

This shows the number of comparison of vectors $(f^1, f^2, ---, f^L)$.

2. D.P. Approach

At each stage, there are (Kn+1) x values and for each value of x there are (x+1) D values. The number of evaluation of objective functions

at one stage is

$$L \sum_{x=0}^{Kn} (x+1) = L((Kn+1) + \frac{Kn(Kn+1)}{2} .$$

All (N-1) stages have the same number of enumerations

$$L \times (N-1)\left[Kn+1) + \frac{Kn(Kn+1)}{2}\right]$$

and last stage has just (Kn+1) x L computations. Thus

$$\left[(N-1)((Kn+1) + \frac{Kn(Kn+1)}{2}) + (Kn+1)\right] \times L$$

and the number of comparisons for choosing the nondominated solutions is at each stage

$$\sum_{x=1}^{Kn} \frac{x(x+1)}{2} .$$

and since all (N-1) stages are the same, the total number is

$$(N-1) \sum_{x=1}^{Kn} \frac{x(x+1)}{2} .$$

At the last stage, the number of comparisons is $\frac{Kn(Kn+1)}{2}$

thus the total number in N-1 stage is

$$(N-1) \sum_{x=1}^{Kn} \frac{x(x+1)}{2} + \frac{Kn(Kn+1)}{2} .$$

Therefore total number of comparisons for all stages is

$$(N-1)\left[\frac{Kn}{12}(Kn+1)(2Kn+1) + \frac{Kn(Kn+1)}{4} + \frac{Kn(Kn+1)}{2} .\right.$$

For example (2 objective functions)

DIRECT ENUMERATION

N \ K_n	10	50	100	1000
3	572	4.7×10^4	3.5×10^5	3.3×10^8
5	6×10^3	6.9×10^6	1.9×10^8	1.7×10^{13}
10	3.7×10^5	1.5×10^{11}	9.2×10^{13}	5.8×10^{23}

DYNAMIC PROGRAMMING

N \ K_n	10	50	100	1000
3	286	5.4×10^3	2.1×10^4	2×10^6
5	550	1×10^4	4.1×10^4	4×10^6
10	1210	2.4×10^4	9.2×10^4	9×10^6

Notice that the direct enumerations grow exponentially and the dynamic programming approach increases linearly with N.

IV. CONCLUSIONS

We have discussed the concept of the nondominated solutions and have shown how the concept of the dynamic programming can be applied to determine the nondominated point set.

We have also discussed the sufficient conditions for the decomposition of multiple criteria functions. Problems involving both the discrete and continuous case can be handled.

LIST OF REFERENCES

[1] Geoffrion, A. M., "Proper Efficiency and the Theory of Vector Maximization," Journal of Mathematical Analysis and Applications, 22, (1968), pp. 618-630.

[2] MacCrimmon, Kenneth R., "An Overview of Multiple-Objective Decision Making," Multiple Criteria Decision Making edited by James L. Cochrane and Milan Zeleny.

[3] Slovic, P., and S. Lichtenstein, "Comparison of Bayesian and Regression Approaches to the Study of Information Processing in Judgment," Organizational Behavior and Human Performance, 6, 1971, pp. 651-730

[4] Hoffman, P. J., P. Slovic, and L. G. Roper, "An Analysis of Variance Model for the Assessment of Configural Cue Utilization in Clinical Judgment," Psychology Bulletin, 69, 1968, pp. 338-349.

[5] Raiffa, H., "Preference for Multi-attributed Alternatives," (Santa Monica, California; RAND, RH-5868, 1969).

[6] Stouffer, S. A., et al., "The American Soldier: Combat and Its Aftermath," Studies in Social Psychology in World War II, Vol. II, (Princeton, New Jersey: Princeton University Press, 1949).

[7] Terry, H., "Comparative Evaluation of Performance Using Multiple Criteria," Management Science, 9, 1968, pp. 431-441.

[8] Freimer, M., and P. L. Yu, "An Approach Toward Decision Problems with Multiobjectives", Center for System Science, Paper 72-03, University of Rochester, 1972.

[9] Stigler, G. J., "The Cost of Subsistence," Journal of Farm Economics, 27, 1945.

[10] Lee, S., and E. R. Clayton, "A Goal Programming Model for Academic Resource Allocation," Management Science, 18, 1972, pp. B305-B417.

[11] Ijiru, Y., "Management Goal and Accounting for Control," (Amsterdam, North Holland, 1965).

[12] Geoffrion, A., J. S. Dyer, and A. Feinberg, "An Interactive Approach for Multi-criterion Optimization with an Application to the Operation of an Academic Department," Western Management Science Institute, Working Paper 176, University of California, Los Angeles, California, 1971.

[13] Benayoun, R., J. de Montgoltier, and J. Tergny, "Linear Programming With Multiple Objective Functions: Step Method (STEM)," Sema, Paris, 1970.

[14] MacCrimmon, K. R., "Improving the System Design and Evaluation Process by the Use of Trade-off Information," (Santa Monica, California: RAND, RN-5877DOT, 1969).

[15] Roy, D., "Problems and Methods With Multiple Objective Functions," Mathematical Programming 1 (1971), North Holland Publishing Company, pp. 239-266.

[16] Johnsen, Earl, "Multiobjective Management Process, Multiple Criteria Decision Making," (Kyoto, 1975, Springer-Verlag, Berlin, Heidelberg, New York).

[17] Yu, P. L., "Introduction to Domination Structures in Multicriteria Decision Problems," Multiple Criteria Decision Making edited by James L. Cochrane and Milan Zeleny.

[18] Zeleny, Milan, "Compromise Programming," Multiple Criteria Decision Making edited by James L. Cochrane and Milan Zeleny.

[19] Polak, Elijah, "An Algorithm for Bicriteria Optimization Based on the Sensitivity Function," IEEG Transactions on Automatic Control, August 1975.

[20] Bellman, Richard,"Dynamic Programming," Princeton University Press, Princeton, New Jersey, 1957.

[21] Hadley, G., "Nonlinear and Dynamic Programming," Addison-Wesley, Reading, Massachusetts, 1964.

[22] Cochrane, James L., and Milan Zeleny, "Multiple Criteria Decision Making," University of South Carolina Press, Columbia, South Carolina, 1973.

[23] Holl, Stephen Trygue, "Efficient Solutions to a Multicriteria Linear Program, with Application to an Institution of Higher Education," A Dissertation Submitted to the Johns Hopkins University in Informity with the Requirements for the Degree of Doctor of Philosophy, Baltimore, Maryland, 1973.

[24] Zeleny, M., "Multiple Criteria Decision Making," Kyoto, 1975, Springer-Verlog, Berlin, Heidelberg, New York.

[25] Keeney, Ralph L., and Howard Raiffa, "Decision with Multiple Objectives: Preferences and Value Trade Off," 1976, by John Wiley and Sons, Inc.

[26] Roy, B., "Problems and Methods with Multiple Objective Functions," Mathematical Programming 1 (1971), (North Holland Publishing Company), pp. 239-266.

[27] Briskin, Lawrence E., "A Method of Unifying Multiple Objective Functions," Management Science, Vol. 12, No. 10, June 1966.

[28] Terry, Herbert, "Comparative Evaluation of Performance Using Multiple Criteria," Management Science, Vol. 9, No. 3 (1963).

[29] Charnes, A., W. W. Cooper, R. J. Niehaus, and Stedry, "Static and Dynamic Assignment Models with Multiple Objective and Some Remarks on Organization Design," Management Science, Vol. 15, No. 8, April 1969.

[30] Philip, John, "Algorithms for the Vector Maximization Problem," Management Programming 2 (1972), North Holland Publishing Company, pp. 207-229.

[31] Benayoun, J. de Montgolfier, and J. Tergny, "Linear Programming with Multiple Objective Functions: Step Method (STEM)," Mathematical Programming (1971), North Holland Publishing Company, pp. 366-375.

[32] Mitten, L. G., and G. L. Neuhauser, "Optimization of Multistage Separation Processes by Dynamic Programming," The Canadian Journal of Chemical Engineering, October 1963.

[33] Mitten, L. G., "Composition Principles for Synthesis of Optimal Multistage Process," Operations Research 12, (1964), pp. 610-619.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Library Naval Postgraduate School Monterey, California 93940	2
2.	Associate Professor J. K. Hartman, Code 55Hh Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
3.	Associate Professor G. T. Howard, Code 55Hk Department of Operations Research Naval Postgraduate School Monterey, California 93940	1
4.	Operation Research Section Korean Naval HQ Seoul, Korea	2
5.	Kim Kwang Bog Apt. #405. 10 Dong Dang-San Dong Young-Deung Po Seoul, Korea	2
6.	Defense Documentation Center Cameron Station Arlington, Virginia 22314	2